PTO/SB/21 (12-97)
Approved for use through 9/30/00. OMB 0651-0031
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

AF/IW

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/852,008
Filing Date	May 10, 2001
In re Application of:	Mahesh GIRKAR et al.
Group Art Unit	2177
Examiner Name	Le, D.
Attorney Docket Number	50277-1003
Total Number of Pages in This Submission	
Client Docket Number	OID-2000-207-01

ENCLOSURES ***(check all that apply)***

- ☒ Fee Transmittal Form
  - ☒ Fee Attached
- ☐ Amendment / Response
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Response to Missing Parts/ Incomplete Application
  - ☐ Response to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Assignment Papers *(for an Application)*
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition Routing Slip (PTO/SB/69) and Accompanying Petition
- ☐ To Convert a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Small Entity Statement
- ☐ Request of Refund
- ☐ After Allowance Communication to Group
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☒ Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- ☐ Proprietary Information
- ☐ Status Letter
- ☐ Additional Enclosure(s) *(please identify below):*

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm *or* Individual name	DITTHAVONG & CARLSON, P.C. Stephen C. Carlson, Reg. No. 39929
Signature	
Date	November 15, 2004

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Assistant Commissioner for Patents, Alexandria, VA 22313-1450 on this date:

Type or printed name	Stephen C. Carlson		
Signature		Date	November 15, 2004

PTO/SB/17 (01-03)
Approved for use through 04/30/2003. OMB 0651-0032
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

FEE TRANSMITTAL for FY 2004

Effective 10/01/2003. Patent fees are subject to annual revision.

☐ Applicant Claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 330

Complete if Known	
Application Number	09/852,008
Filing Date	May 10, 2001
First Named Inventor	Girkar, et al.
Examiner Name	Le, D.
Art Unit	2177
Attorney Docket No.	50277-1003

METHOD OF PAYMENT *(check all that apply)*

☐ Check ☒ Credit card ☐ Money Order ☐ Other ☐ None

☐ Deposit Account

Deposit Account Number

Deposit Account Name

The Commissioner is authorized to: *(check all that apply)*

☐ Charge fee(s) indicated below ☐ Credit any overpayments

☐ Charge any additional fee(s) during the pendency of this application

☐ Charge fee(s) indicated below, except for the filing fee to the above-identified deposit account.

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1001	770	2001	385	Utility filing fee	
1002	340	2002	170	Design filing fee	
1003	530	2003	265	Plant filing fee	
1004	770	2004	385	Reissue filing fee	
1005	160	2005	80	Provisional filing fee	

SUBTOTAL (1) ($)

2. EXTRA CLAIM FEES FOR UTILITY AND REISSUE

		Extra Claims		Fee from below		Fee Paid
Total Claims	-20**=		X		=	
Independent Claims	- 3**=		X		=	
Multiple Dependent					=	

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	18	2202	9	Claims in excess of 20
1201	86	2201	43	Independent claims in excess of 3
1203	290	2203	145	Multiple dependent claim, if not paid
1204	86	2204	43	**Reissue independent claims over original patent
1205	18	2205	9	**Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($)

** or number previously paid, if greater; For Reissues, see above

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge – late filing fee or oath	
1052	50	2052	25	Surcharge – late provisional filing fee or cover sheet	
1053	130	1053		Non-English specification	
1812	2,520	1812		For filing a request for *ex parte* reexamination	
1804	920*	1804		Requesting publication of SIR prior to Examiner action	
1805	1,840*	1805		Requesting publication of SIR after Examiner action	
1251	110	2251	55	Extension for reply within first month	
1252	420	2252	210	Extension for reply within second month	
1253	950	2253	475	Extension for reply within third month	
1254	1,480	2254	740	Extension for reply within fourth month	
1255	2,010	2255	1,005	Extension for reply within fifth month	
1401	330	2401	165	Notice of Appeal	
1402	330	2402	165	Filing a brief in support of an appeal	330
1403	290	2403	145	Request for oral hearing	
1451	1,510	1451		Petition to institute a public use proceeding	
1452	110	2452	55	Petition to revive – unavoidable	
1453	1,330	2453	665	Petition to revive – unintentional	
1501	1,330	2501	665	Utility issue fee (or reissue)	
1502	480	2502	240	Design issue fee	
1503	640	2503	320	Plant issue fee	
1460	130	1460		Petitions to the Commissioner	
1807	50	1807		Processing fee under 37 CFR 1.17(q)	
1806	180	1806		Submission of Information Disclosure Stmt	
8021	40	8021	40	Recording each patent assignment per property (times number of properties)	
1809	770	2809	385	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	770	2810	385	For each additional invention to be examined (37 CFR § 1.129(b))	
1801	770	2801	385	Request for Continued Examination (RCE)	
1802	900	1802		Request for expedited examination of a design application	

Other fee (specify)

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($) 330

SUBMITTED BY

Complete *(if applicable)*

Name (Print/Type)	Stephen C. Carlson	Registration No. (Attorney/Agent)	39929	Telephone	703-425-8516
Signature				Date	November 15, 2004

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	Mahesh GIRKAR et al.		
Application No.:	09/852,008	Group Art Unit:	2177
Filed:	May 10, 2001	Examiner:	Le, D.
Attorney Docket:	50277-1003		
Client Docket:	OID-2000-207-01		

For: DISASTER RECOVERY WITH BOUNDED DATA LOSS

APPEAL BRIEF

Honorable Commissioner for Patents
Alexandria, VA 22313-1450

Dear Sir:

This Appeal Brief is submitted in support of the Notice of Appeal dated September 10, 2004.

I. REAL PARTY IN INTEREST

Oracle International Corp. is the real party in interest.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals and interferences.

III. STATUS OF THE CLAIMS

Claims 1-16 are pending in this appeal, in which claims XXX have earlier been canceled. No claim is allowed. This appeal is therefore taken from the final rejection of claims 1-16 on March 16, 2004.

IV. STATUS OF AMENDMENTS

No amendment to claims has been filed after final rejection.

V. SUMMARY OF THE INVENTION

The present invention addresses problems addresses difficulties in database systems. For example, the Background section of the present application explains that "it is difficult to characterize the amount of data lost in terms that database owners can best understand. The maximum exposure for loss of data in this approach is usually described in terms of the size of the redo logs, but this information is not helpful for database owners, who would rather want to know how many orders were lost" (Background, ¶ 7). In recognition of this problem, the method recited in the claims sets forth the use of "a predetermined number of transactions" in synchronizing a transaction (claim 1; see FIG. 3, item 303 showing the use of a transaction counter). "Because the predetermined bound is specified in terms of the number of transactions, the database operator can set a meaningful tradeoff between performance and data availability that is appropriate for the particular needs of the database operator's installation" (Summary, ¶ 11).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether claims 1-6, 8, and 10 are obvious under 35 U.S.C. § 103(a) based on *Rastogi et al.* (U.S. 6,205,449) in view of *Cooper et al.* (U.S. 6,079,000)?

Whether claims 7 and 9-15 are obvious under 35 U.S.C. § 103(a) based on *Rastogi et al.* in view of *Cooper et al.* and further in view of *Hapner et al.* (U.S. 5,940,827)?

Whether claim 14 is obvious under 35 U.S.C. § 103(a) based on *Rastogi et al.* in view of *Hapner et al.*?

Whether claim 16 is obvious under 35 U.S.C. § 103(a) based on *Rastogi et al.* in view of *Cooper et al.* and further in view of *Nilsen et al.* (U.S. 5,668,986).

VII. ARGUMENT

A. ALL PENDING CLAIMS ARE NON-OBVIOUS OVER *RASTOGI ET AL.* BECAUSE NONE OF THE REFERENCES TEACH OR SUGGEST SYNCHRONIZING TRANSACTIONS BASED ON A "PREDETERMINED NUMBER OF TRANSACTIONS."

The initial burden of establishing a *prima facie* basis to deny patentability to a claimed invention under any statutory provision always rests upon the Examiner. *In re Mayne*, 104 F.3d 1339, 41 USPQ2d 1451 (Fed .Cir. 1997); *In re Deuel*, 51 F.3d 1552, 34 USPQ2d 1210 (Fed. Cir. 1995); *In re Bell*, 991 F.2d 781, 26 USPQ2d 1529 (Fed. Cir. 1993); *In re Oetiker*, 977 F.2d 1443, 24 USPQ2d 1443 (Fed. Cir. 1992). In rejecting a claim under 35 U.S.C. § 103, the Examiner is required to provide a factual basis to support the obviousness conclusion. *In re Warner*, 379 F.2d 1011, 154 USPQ 173 (CCPA 1967); *In re Lunsford*, 357 F.2d 385, 148 USPQ 721 (CCPA 1966); *In re Freed*, 425 F.2d 785, 165 USPQ 570 (CCPA 1970).

1. CLAIMS 1-10 ARE NON-OBVIOUS OVER *RASTOGI ET AL.* IN VIEW OF *COOPER ET AL.*

Reversal of the rejection of claims 1-10 over *Rastogi et al.* and *Cooper et al.* (with or without *Hapner et al.*) is respectfully requested because the references do not teach or otherwise suggest the limitations of the claims. For example, independent claim 1 recites: "synchronizing a transaction performed on the primary database system based on a number of transactions in the buffer and **a predetermined number of transactions**."

The Examiner correctly acknowledges that "Rastogi does not explicitly teach a predetermined number of transactions" (p. 3). Indeed, the portion of *Rastogi et al.* cited for transactions, col. 8:3-8, only discusses "transactions" in the model described in the reference. There is no mention or suggestion of "synchronizing a transaction," much less "synchronizing a transaction" based on "a predetermined number of transactions."

Cooper et al. too fails to show this feature. In fact, *Cooper et al.* exhibits many of the same difficulties described in the background and addressed by the invention recited in claim 1. For example, *Cooper et al.* recommends managing the "optimal transfer efficiency" between the audit host memory **342** and the XPC cache area **350** based on a "predetermined size of audit host memory **342**," which is given in terms of a "predetermined number of address locations within audit host memory **342**" (col. 12:39-40). Referring now to FIG. 9 of *Cooper et al.*, transactions **344**, **354**, **362**, and **370** clearly have different sizes in terms of the number of audit host memory **342** locations. For example, transaction **354** is about half the size of transaction **344** and about a third of the size of transaction **362**. When transaction sizes are so variable as in *Cooper et al.*, pre-specifying buffers in terms of number of memory locations or bytes does not meaningfully specify "a predetermined number of transactions" as set forth in independent claim 1. In fact, this

variability in transaction size is what dooms *Cooper et al.*'s approach to exhibit the problems addressed by the invention of independent claim 1.

The portion of *Cooper et al.* cited in the Office Action, col. 12:30-43, does not support the rejection. Although *Cooper et al.* speculates that "the optimal transfer characteristics of the physical audit trail **378** may determine when a sufficient number of transactions have been accumulated in audit host memory **342**" (col. 12:33-36), *Cooper et al.* then defines what it means a "sufficient number of transactions"—not in terms of a "predetermined number of transactions" as recited in claim 1—but explicitly in terms of the size of audit host memory **342**: "Thus, the optimal transfer efficiency may correspond to the a predetermined size of audit host memory **342**" (col. 12:36-38). Due to the variability in transaction sizes evident in FIG. 9, *Cooper et al.*'s criterion based on a predetermined memory size can only crudely and inaccurately correspond to the actual number of transactions in the audit host memory **342**. However, *Cooper et al.* is not concerned with providing a meaningful bound to database administrators but focused on transferring data between memories as efficiently as possible. Thus, *Cooper et al.* actually **teaches against** using a "predetermined number of transactions," since transferring one predetermined number of tiny transactions is less efficient than transferring the same predetermined number of large transactions. If a proposed modification would render the reference being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984).

As for col. 14:41-43, also cited in the Office Action, *Cooper et al.* makes a similar recommendation for the size of the XPC cache area **350** in terms of "a predetermined number of address locations within XPC cache area **350**" (col. 14:39-41) and is therefore similarly deficient for the same reasons as with the size of audit host memory **342**. Furthermore, *Cooper et al.* states

that the "synchronous audit data request results in a portion of the audit host memory **342** being written to a corresponding portion of non-volatile memory storage such as XPC cache **350**" (col. 11:50-53), thus the relevance of the size of the XPC cache area **350** to "synchronizing a transaction" as recited in independent claim 1 is not immediately apparent.

2. CLAIMS 7, 9, AND 11-16 ARE NON-OBVIOUS OVER *RASTOGI ET AL.* IN VIEW OF *HAPNER ET AL.*

Concerning dependent claims 7 and 9 as well as claims 11-16, the use of the non-analogous *Hapner et al.* does not support the rejection by curing the deficiencies of *Rastogi et al.* and *Cooper et al.* or by disclosing the additional features recited in claims 7, 9, and 11-16. *Hapner et al.* relates to maintaining a database cache in conjunction with a persistent database portion and uses a "transaction counter" (col. 3:44) for keeping track of how many open transactions there are in the database cache **140**. *Hapner et al.* lacks any disclosure of using such a "transaction counter" for any set of "transactions to be sent to a standby database system." Since neither *Rastogi et al.* nor *Cooper et al.* operate with a predetermined number of transactions to be sent to a standby database system, there is no motivation to modify *Rastogi et al.* and/or *Cooper et al.* to count something none of the references seem to care about.

Furthermore, the only comparison of *Hapner et al.*'s transaction counter is with zero (0) in FIG. 10, step 469, and FIG. 11, steps 512 and 536. However, claim 11 explicitly recites "**compare** the counter and **the predetermined bound**" and claim 12 recites "**comparing** a counter indicating a number of the transactions in a queue of transactions to be sent to a standby database system and **a predetermined bound of transactions**." Whatever *Cooper et al.* may be thought to disclose about the optimal transfer size of audit host memory **342**, the optimal transfer size certainly cannot be zero! Thus, even if the Examiner might be correct in responding that

"Hapner does apply a transaction counter in a replicating data" (p. 11), the claims are more specific than that and the use of a counter in the specific context of claims 7, 9, and 11-16 is not disclosed in *Hapner et al.*

3. CLAIM 16 IS NON-OBVIOUS OVER *RASTOGI ET AL.* FURTHER IN VIEW OF *NILSEN ET AL.*

Nilsen et al., applied only against claim 16, does not furnish a disclosure of "synchronizing a transaction performed on the primary database system based on a number of transactions in the buffer and the corresponding bound" which is missing in *Rastogi* and *Cooper et al.* The Examiner, properly, did not rely not *Nilsen et al.* for this factual inadequacy of *Rastogi et al.* and *Cooper et al.* Thus, claim 16 too is patentable over *Rostogi et al.*, *Cooper et al.*, and *Nilsen et al.*, and its rejection is respectfully requested.

VIII. CONCLUSION AND PRAYER FOR RELIEF

For the foregoing reasons, Appellants request the Honorable Board to reverse each of the Examiner's rejections.

Respectfully Submitted,

DITTHAVONG & CARLSON, P.C.

11/15/2004
Date

Stephen C. Carlson
Attorney for Applicant(s)
Reg. No. 39929

10507 Braddock Rd, Suite A
Fairfax, VA 22032
Tel. 703-425-8516
Fax. 703-425-8518

APPENDIX

1. (Original) A method for replicating data of a primary database system, comprising the steps of:

maintaining a buffer of transactions to be sent to a standby database system; and

synchronizing a transaction performed on the primary database system based on a number of transactions in the buffer and a predetermined number of transactions.

2. (Original) A method according to claim 1, wherein the step of synchronizing includes the step of:

blocking a commit of the transaction until the number of transactions in the buffers is in a predetermined numerical relationship with the predetermined number of transactions.

3. (Previously Presented) A method according to claim 2, wherein:

said blocking the commit of the transaction until the number of transactions in the buffers is in the predetermined numerical relationship with the predetermined number of transactions including blocking the commit of the transaction until the number of transactions in the buffers is less than the predetermined number of transactions.

4. (Original) A method according to claim 1, further comprising the step of:

executing a log writer process to record the transaction in a redo log.

5. (Original) A method according to claim 4, wherein:

the log writer process performs the step of synchronizing.

6. (Original) A method according to claim 4, wherein:

a database application process performs the step of synchronizing before submitting the transaction to the log writer process.

7. (Original) A method according to claim 1, further comprising the step of:

executing a net server process to:

transmit the transaction over a network connection to the standby database system,

receive an acknowledgment that a redo record for the transaction has been written to a standby log at the standby database system, and

remove the transaction from the buffer in response to the acknowledgment.

8. (Original) A method according to claim 1, further comprising the steps of:

receiving input from an operator indicating a transaction loss bound; and

setting the predetermined number of transactions based on the transaction loss bound.

9. (Original) A method according to claim 1, wherein the step of synchronizing includes the steps:

storing a counter indicating a number of the transactions in the buffer;

when adding the transaction to the buffer, incrementing the counter;

when removing the transaction from the buffer, decrementing the counter;

blocking a commit of the transaction when the counter is not less than the predetermined number of transactions; and

acknowledging the commit of the transaction when the counter is less than the predetermined number of transactions.

10. (Original) A computer-readable medium bearing instructions for causing one or more processors to performs the steps of the method according to claim 1.

11. (Original) A method for replicating data of a primary database system, comprising the steps of:

maintaining a queue of transactions to be sent to a standby database system;

storing a counter indicating a number of the transactions in the queue;

storing a predetermined bound of transactions;

executing a log writer process to:

record the transaction in a redo log,

compare the counter and the predetermined bound,

if the counter is not less than the predetermined bound, then block a commit of the transaction until the counter is less than the predetermined bound, and

if the counter is less than the predetermined bound, then increment the counter and acknowledge the commit of the transaction; and

executing a net server process to:

transmit the transaction over a network connection to the standby database system,

receive an acknowledgment that a redo record for the transaction has been written to a standby log at the standby database system, and

in response to the acknowledgment, remove the transaction from the queue and decrement the counter.

12. (Original) A method for operating a log writer process to replicate data of a primary database system, comprising the steps of:

recording a transaction in a redo log;

comparing a counter indicating a number of the transactions in a queue of transactions to be sent to a standby database system and a predetermined bound of transactions;

if the counter is not less than the predetermined bound, then blocking a commit of the transaction until the counter is less than the predetermined bound, and

if the counter is less than the predetermined bound, then incrementing the counter and acknowledging the commit of the transaction.

13. (Original) A computer-readable medium bearing instructions for causing one or more processors to perform the steps of the method according to claim 12.

14. (Previously Presented) A method for operating a net server process to replicate data of a primary database system, comprising the steps of:

accessing a transaction maintained in a buffer of transactions to be sent to a standby database system;

transmitting the transaction over a network connection to the standby database system;

receiving an acknowledgment that a redo record for the transaction has been written to a standby log at the standby database system, and

in response to the acknowledgment, removing the transaction from the queue and decrementing a counter.

15. (Original) A computer-readable medium bearing instructions for causing one or more processors to perform the steps of the method according to claim 14.

16. (Original) A method for replicating data in a primary database system having multiple database servers operating in parallel and accessing a common database on a shared disk, said method comprising the steps of:

setting a bound for each of the multiple database servers;

for each of the multiple database servers, performing the steps of:

maintaining a buffer of transactions to be sent to a standby database system; and

synchronizing a transaction performed on the primary database system based on a number of transactions in the buffer and the corresponding bound.